

### **Remarks**

This Amendment responds to the final Office Action (“the Action”) mailed August 29, 2006. Reconsideration of the application is respectfully requested in view of the foregoing amendments and following remarks. Claims 1-20 are pending in the application. No claims have been allowed. Claims 1, 10, and 16 are independent.

### ***Cited Art***

U.S. Patent No. 4,989,132 to Mellender et al. ("Melender") is entitled “Object-Oriented, Logic, and Database Programming Tool with Garbage Collection.”

### ***Amendments***

Claims 1, 10, and 16 have been amended to more clearly demonstrate that they recite statutory subject matter and to correct typographical errors. The amendments are not made in response to rejections under 35 U.S.C. § 102 over Mellender. No new matter has been added.

### ***Rejection Under § 101***

The Action rejects claims 1-20 under 35 U.S.C. § 101 as being directed to non-statutory subject matter. Applicants respectfully disagree with the rejection of claims 1-20. However, in order to expedite examination of the claims, claims 1, 10, and 16 have each been amended to more clearly demonstrate their statutory nature.

The Action specifically alleges that independent claim 1 does not produce a “concrete, tangible, and useful result” and that the claim body “lacks functional descriptive material consisting of data structures and programs supporting or allowing a computer to execute to produce a concrete, tangible, and useful result.”

Claim 1 has been amended as follows:

A computer system for managing memory for an executing object-oriented program, the program comprising a plurality of objects and the system comprising:

- a processing unit;*
- a computer-readable memory partitioned into a plurality of regions, each region containing at least one object *in the memory*; and
- at least one *data structure in the computer-readable memory representing at least one region shape-graph* describing relations between the regions, the

relations based at least in part on references between objects contained in the regions;

*wherein the processing unit is configured such that given a reference to a target object, the processing unit can identify the partitioned region of memory containing the target object by using information from the at least one region shape graph.*

[emphasis added.] Thus, claim 1 now clearly recites a “data structure” resident in memory which represents the “shape graph” as well a processing unit which is configured to “identify the partitioned region of memory containing [a] target object.” As such, claim 1 now more clearly recites statutory subject matter.

Claim 10 was also rejected under 35 U.S.C. § 101 for not producing a “concrete, tangible, and useful result.” Specifically, the Action alleged that the method recited in claim 10 is abstract because “compiling the program by itself and without a further performing step(s), . . . does not produce a concrete, tangible and useful result.” Claim 10 as amended now recites, however,

compiling the program to produce executable code for the program; and  
storing the executable code in a computer-readable storage medium.

As such, claim 10 now more clearly recites statutory subject matter.

Claim 16 was rejected under 35 U.S.C. § 101 for allegedly covering wireless telecommunications signals and carrier waves, which are forms of energy, under the recited language “computer-readable medium.” As the Action alleges, these signals are not matter, a composition of matter, or a product. Claim 16, as amended, now recites a “computer-readable storage medium.” As such, claim 16 now more clearly recites statutory subject matter.

Claims 1, 10, and 16 should now be allowable under 35 U.S.C. § 101. Additionally, dependent claims 2-9, 11-15, and 17-20, which the Action rejects or objects to as failing to remedy the alleged deficiencies of the independent claims, should be allowable as well. In particular, claims 12-15 and 17-20, which the Action indicates as allowable except for objection under § 101, are allowable at this time. Applicants thus request allowance of claims 1-20.

***Patentability of Claims 1-11 and 16 over Melender under § 102(b)***

The Action rejects claims 1-11 and 16 under 35 U.S.C. § 102(e) as being anticipated by Mellender. Applicants respectfully disagree and traverse the rejection. For a 102(b) rejection to be proper, the cited art must show each and every element as set forth in a claim. (See MPEP § 2131.01.) However, the cited art does not describe each and every element. Accordingly, applicants request that the rejection be withdrawn. Applicants also note that the arguments presented herein in response to the rejections of claims 1-11 and 16 are not based on any of the amendments made herein, which were done for the purpose of responding to the § 101 rejections discussed above.

*Claim 1*

Claim 1, as amended, recites, in part:

a computer-readable memory partitioned into a plurality of regions . . . ;  
and  
at least one data structure in the computer-readable memory representing  
*at least one region shape graph describing relations between the regions, the relations based at least in part on references between objects contained in the regions; . . .*

[emphasis added.] The application provides examples of shape graphs:

The shape graph 250 contains metadata which maintains associations between regions based on object references by containing nodes representing regions and directed edges between the nodes which demonstrate associations between regions. In one implementation, the shape graph edges represent reference names. As an example, the shape graph 250 could contain information that any object referenced by a field called “age” in an object in region  $\alpha$  will be contained in region  $\beta$  by containing  $\alpha$  and  $\beta$  nodes connected by an “age” edge. In alternate implementations, shape graphs may associate regions by different or additional information, such as object type, level of protection of the reference, or by utilizing unique field identifiers. In one implementation shape graphs are maintained as data structures describing nodes and edges. In alternate implementations, different data structures may be used, as long as the structures comprise metadata which maintains associations between regions.

[Application, at page 6, line 23 to page 7, line 4.] In its rejection of claim 1, Action refers to the “regions” which are described in Figures 17 and 18 and in column 60 of Mellender.

*The region numbers of Mellender do not teach or suggest the “region shape graph describing relations between the regions, the relations based at least in part on references between objects contained in the regions” as recited in claim 1, because the Mellender’s region*

*numbers are a simple count which does not represent relations between regions.* Mellender describes the region numbers at Figures 17 and 18 at in the specification:

When a context is created, it is assigned a region number. Once assigned, a context's region number never changes. Each object created or accessed is assigned the region number of the context that created or accessed it, unless it was already associated with a region with a lower number. After the number of objects in the 'current' region exceeds a fixed maximum, a new region (with an id one greater than the previous one) is started when the next context is created. Thus the region number is the same or increases as one travels down the context stack from sender to receiver.

[Mellender, at column 60, lines 24-34.] As this passage demonstrates, the region numbers it describes are simply a count which increases for each new context created in Mellender's Alltalk system. Mellender supports this count notion further in its discussion of garbage collection based on the region numbers:

Most garbage objects are collected by the collector 200, using the following logic. When returning from a method, if the context to which the process is returning belongs to a region with an id at least two lower than the current region number before returning, the regions with id higher than that of the context to which it is returning are collected. Referring to the Drawings, FIG. 22 shows, in case 1, a context in region n returning to another in region n. Since the region number is the same, no action is taken. Case 2 shows a context in region n+1 returning to one in region n. Since n+1 is not two larger than n, no action is taken. Case 3 shows a context in region n+2 returning to one in region n. Since n+2 is two larger than n, the collector 200 collects regions n+2 and n+1, and all other regions having number greater than n.

[Mellender, at column 62, lines 28-43.] Thus, even in the context of garbage collecting, relations between the regions (based on relations between objects in the regions) are not recorded, let alone kept in a shape graph, because the garbage collector is interested in deleting objects based only on a numerical order provided by the count of the region numbers.

For at least these reasons, Mellender does not teach or suggest the above-recited language of claim 1 and thus Mellender does not describe each and every element of claim 1. Claim 1, as well as claims 2-9, which depend from claim 1, are thus allowable and applicants request their allowance.

*Claim 10*

Claim 10, as amended, recites, in part:

*performing a points-to analysis on the source code to develop at least one data structure containing region association metadata* for the program;

[emphasis added.] The application describes processes for creating shape graphs which contain the “region association metadata,” recited in claim 16, for example, at Figure 4 and page 8, line 9 to page 10, line 19. At Figure 3, and page 7, line 25 to page 8, line 8, the Application also demonstrates that this process can be performed independently of compilation of a program into executable. See, for example, the separate blocks 310 and 330 of Figure 3, which illustrate the “analyze program code to create shape graphs” and “generate execution code” actions, respectfully.

*The compilation process described in Mellender does not teach or suggest the “points-to analysis on the source code to develop at least one data structure containing region association metadata” language of claim 1, as it describes standard compiler parsing.* In its rejection of the “point-to analysis” language claim 10, the Action cites to column 6, line 29 through column 7, line 14 of Mellender. However, this section discusses the creation of a parse tree when compiling the Alltalk language of Mellender. At no point in this discussion does the passage discuss an analysis “to develop at least one data structure containing region association metadata” for a program. And while the Action also cites column 2, lines 40-44 in this rejection, this passage describes only a *database* storage system for objects:

The database has a key file and a prime file. The prime file contains records of variable length for storing objects, and the key file contains records of fixed length for storing the address, record length, and type of object in the prime file.

[Mellender, at column 2, lines 39-44.] Applicants further do not find a connection between the cited parse tree creation process of columns 6 and 7 and the prime file database format of column 2, nor do Applicants find further discussion of the recited “points-to analysis” elsewhere in Mellender. For at least these reasons, Mellender does not teach or suggest the above-recited language of claim 10 and thus Mellender does not describe each and every element of claim 10. Claim 10, as well as claim 11, which depends from claim 1, are thus allowable and applicants request their allowance.

*Claim 16*

Claim 16, as amended, recites, in part:

performing a points-to analysis on the source code to develop at least one shape-graph template for the program;

In its rejection of claim 16, the Action cites to the same passages of Mellender as it does in its rejection of claim 10. Thus, for at least the reasons discussed above with respect to claim 10, Mellender does not teach or suggest the above-recited language of claim 16 and thus Mellender does not describe each and every element of claim 16. Claim 16 is thus allowable and applicants request its allowance:

***Claim Amendments Do Not Provide Grounds for Finality in Next Office Action***

Applicants respectfully note that the arguments made above with respect to the rejections under 35 U.S.C. § 102 did not rely on any amendments made in this office action. Indeed, the amendments were made solely to demonstrate that the claims covered statutory subject matter under 35 U.S.C. § 101. Thus, applicants respectfully submit that, should the arguments above succeed and the Examiner introduce a new ground of rejection, that rejection would not be “necessitated by applicant's amendment of the claims” and thus not fulfill the requirements for finality as required by M.P.E.P. § 706.07(a). If the Examiner disagrees with this assertion, Applicants formally request the Examiner to contact the attorney signed below.

***Request for Interview***

If any issues remain, the Examiner is formally requested to contact the undersigned attorney prior to issuance of the next Office Action in order to arrange a telephonic interview. It is believed that a brief discussion of the merits of the present application may expedite prosecution. Applicants submit the foregoing formal Amendment so that the Examiner may fully evaluate Applicants' position, thereby enabling the interview to be more focused.

This request is being submitted under MPEP § 713.01, which indicates that an interview may be arranged in advance by a written request.

***Conclusion***

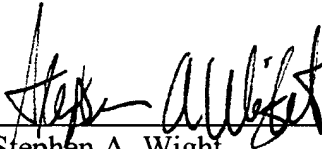
The claims in their present form should now be allowable. Such action is respectfully requested.

Respectfully submitted,

KLARQUIST SPARKMAN, LLP

One World Trade Center, Suite 1600  
121 S.W. Salmon Street  
Portland, Oregon 97204  
Telephone: (503) 595-5300  
Facsimile: (503) 595-5301

By



Stephen A. Wight

Registration No. 37,759